ARK

SDP18 | Team 21

FPR

4/11/2018

Department of Electrical and Computer Engineering

Advisor: Professor Holcomb

Team

Matteo Bolognese, EE
Hardware Guru
Device Design, Assembly
PCB Design, & Power

Chad Klinefelter, CSE
Backend Guru
Data Management &
App Development

Jackie Lagasse, CSE
UI/UX Guru
Augmented Reality &
App Development

Ethan Miller, EE
Algorithms Guru
Device Communication &
Sensor Interfacing

Problem Statement

- AR users are unable to incorporate extremities into an interactive application unless their device camera can directly face them
- There are no inertial sensing products for extremity tracking that can be used in AR applications

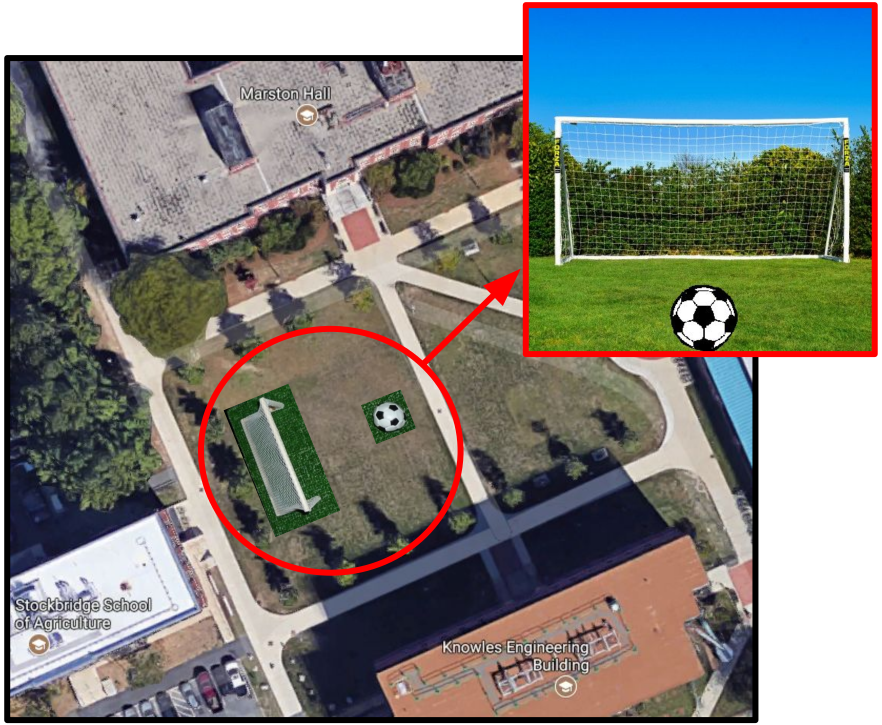
Solution: a separate foot attachment

- Sensor on foot can transmit data to phone, where it can then be acted upon

Our Vision

On demo day we plan to present the following:

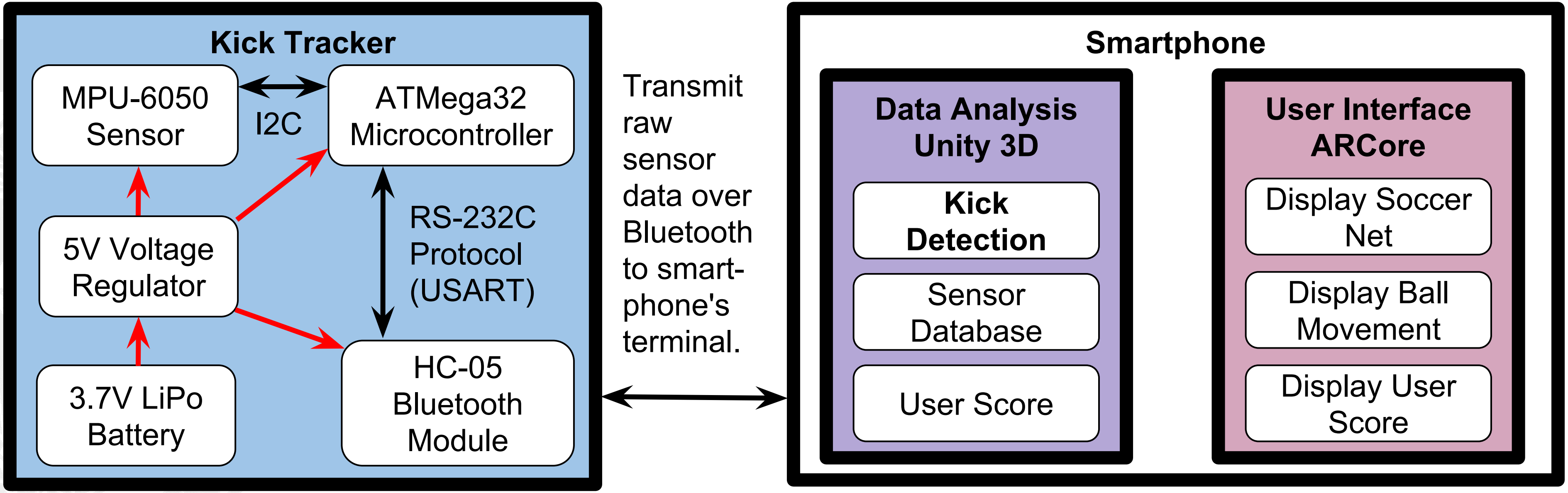
- Bring user to an open space
- User wears headset and kick tracker
- Start ARK app on phone
- App displays virtual soccer ball and goal
- User kicks foot, observes movement of soccer ball


Marston Hall
Stockbridge School of Agriculture
Knowles Engineering Building

System Requirements

Requirement	Status
App connects to Bluetooth and begins game in less than 10 seconds	Met - 2-5 seconds on average
App must determine user’s kick speed and direction	Met - Straight, Angle, 90 deg kicks implemented
Ball must move with speed and direction proportional to user’s foot	Met - Unity game engine maps input vectors to ball as a rigid body
System delay < 300ms, ideally <100ms	Met - 18.7 Hz from sensor to Unity, 60 Hz refresh, 5 frames = 83ms delay
Maximum dimensions of device: 4 x 3 x 2 inches	Met - 4.5 x 3 x 1.25 - overall volume is within spec
Maximum weight: 1 lb	Met - Weight 4.6oz less than 1/3 of spec)
Minimum battery life: 5 hours	Met - Sources ~65mA during operation = 46hr of battery life

Block Diagram



All components in kick tracker are connected together in PCB.
Battery is rechargeable.

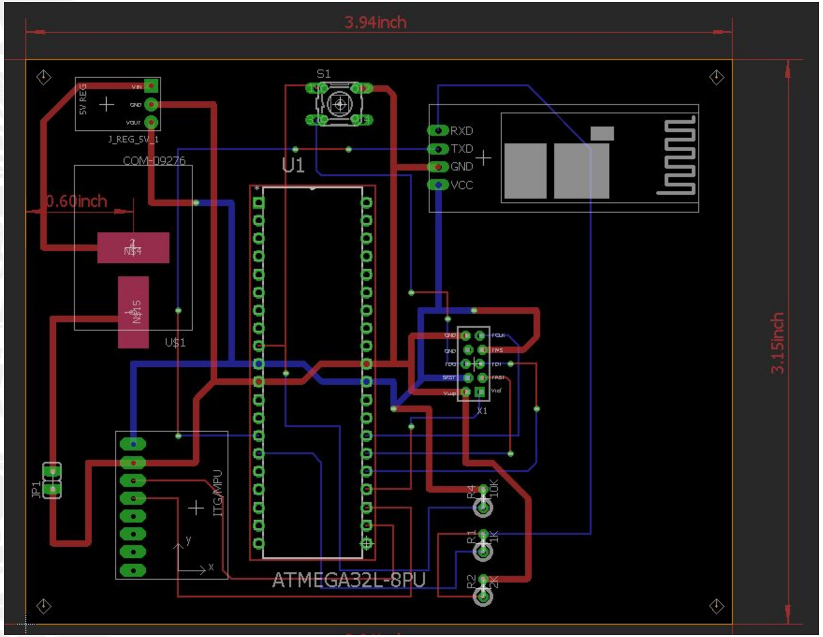
User data consists of user's score in this play through.

FPR Deliverables Status

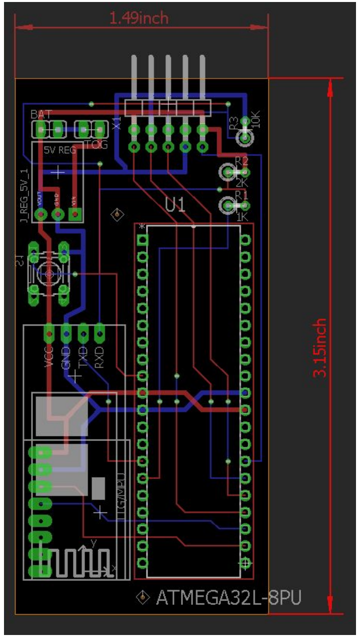
Deliverable	**Member**	**Status**
Hardware: Assemble final case along with new PCB for more efficient layout	Matteo	Completed
Backend: Finish implementing sensor data storage system and refine data storage as kick model is finalized	Chad	Completed
Algorithms: Implement algorithms for at least 2 different kick types and refine thresholds to optimize kick detection for a variety of users	Ethan	Completed
Update UI: Implement left and right foot calibration and improve user interface	Jackie	Completed

PCB Layout

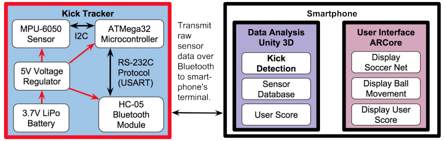
Old



New



- Both PCB designs are complete and work properly
- New PCB layout is almost ⅓ size of old board
- Same width, significantly smaller length
- Better fit for shoes

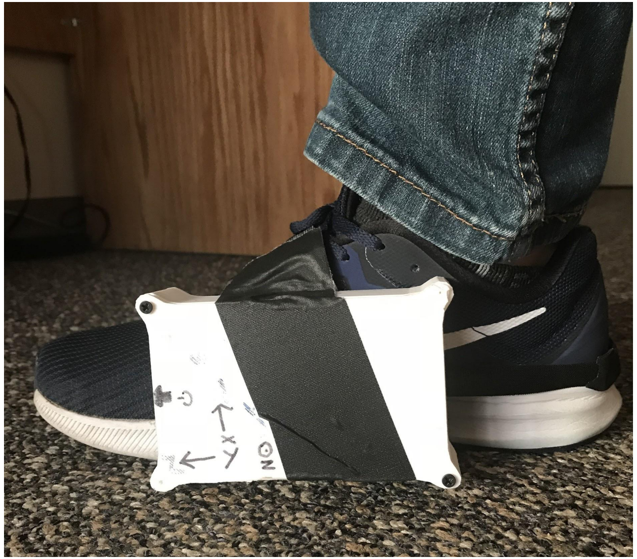


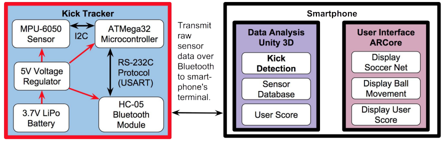
All components in kick tracker are connected together in PCB. Battery is rechargeable.

User data consists of user's score in this play through.

Enclosure

- Current Dimensions:
  - Length - 3”
  - Width - 4.5”
  - Height - 1.25”
  - Thickness - 0.125”
- Material: 3D Solutech (White) Polylactic Acid





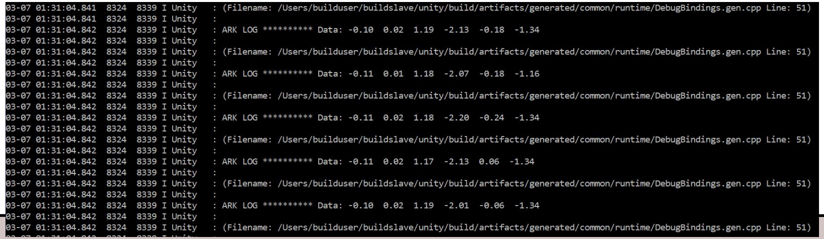
Transmit raw sensor data over Bluetooth to smart-phone's terminal.

All components in kick tracker are connected together in PCB. Battery is rechargeable.

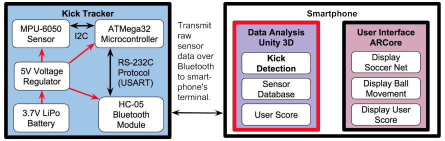
User data consists of user's score in this play through.

Sensor Data / Storage

- Caching recent most ~300 ms (5 samples) of data
  - Get kick peaks with reduced noise by averaging last 5 samples
- **User delay: 83 ms**
  - Data rate: 18.7 Hz (data update every 54 ms)
    - Upon each data update, assume we have 4 previous samples as well
  - Frame rate: 60 Hz (frame update every 17 ms)
  - *Delay:* 4 frames for data arrival + 1 frame for processing = 83 ms



```
03-07 01:31:04.841  8324  8339 I Unity   : (Filename: /Users/builduser/buildslave/unity/build/artifacts/generated/common/runtime/DebugBindings.gen.cpp Line: 51)
03-07 01:31:04.841  8324  8339 I Unity   :
03-07 01:31:04.842  8324  8339 I Unity   : ARK LOG ********** Data: -0.10  0.02  1.19  -2.13  -0.18  -1.34
03-07 01:31:04.842  8324  8339 I Unity   :
03-07 01:31:04.842  8324  8339 I Unity   : (Filename: /Users/builduser/buildslave/unity/build/artifacts/generated/common/runtime/DebugBindings.gen.cpp Line: 51)
03-07 01:31:04.842  8324  8339 I Unity   :
03-07 01:31:04.842  8324  8339 I Unity   : ARK LOG ********** Data: -0.11  0.01  1.18  -2.07  -0.18  -1.16
03-07 01:31:04.842  8324  8339 I Unity   :
03-07 01:31:04.842  8324  8339 I Unity   : (Filename: /Users/builduser/buildslave/unity/build/artifacts/generated/common/runtime/DebugBindings.gen.cpp Line: 51)
03-07 01:31:04.842  8324  8339 I Unity   :
03-07 01:31:04.842  8324  8339 I Unity   : ARK LOG ********** Data: -0.11  0.02  1.18  -2.20  -0.24  -1.34
03-07 01:31:04.842  8324  8339 I Unity   :
03-07 01:31:04.842  8324  8339 I Unity   : (Filename: /Users/builduser/buildslave/unity/build/artifacts/generated/common/runtime/DebugBindings.gen.cpp Line: 51)
03-07 01:31:04.842  8324  8339 I Unity   :
03-07 01:31:04.842  8324  8339 I Unity   : ARK LOG ********** Data: -0.11  0.02  1.17  -2.13  0.06  -1.34
03-07 01:31:04.842  8324  8339 I Unity   :
03-07 01:31:04.842  8324  8339 I Unity   : (Filename: /Users/builduser/buildslave/unity/build/artifacts/generated/common/runtime/DebugBindings.gen.cpp Line: 51)
03-07 01:31:04.842  8324  8339 I Unity   :
03-07 01:31:04.842  8324  8339 I Unity   : ARK LOG ********** Data: -0.10  0.02  1.19  -2.01  -0.06  -1.34
03-07 01:31:04.842  8324  8339 I Unity   :
03-07 01:31:04.842  8324  8339 I Unity   : (Filename: /Users/builduser/buildslave/unity/build/artifacts/generated/common/runtime/DebugBindings.gen.cpp Line: 51)
```

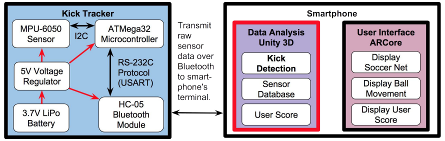


All components in kick tracker are connected together in PCB.
Battery is rechargeable.

User data consists of user's score in this play through.

Kick Detection

- Extracted features from time series via MATLAB
  - Each axis (X,Y,Z) has a defined range for each kick type
  - Sample must concurrently fall within each axis' range for kick to register
- Implemented in-game via a conditional assignment (easy to process)

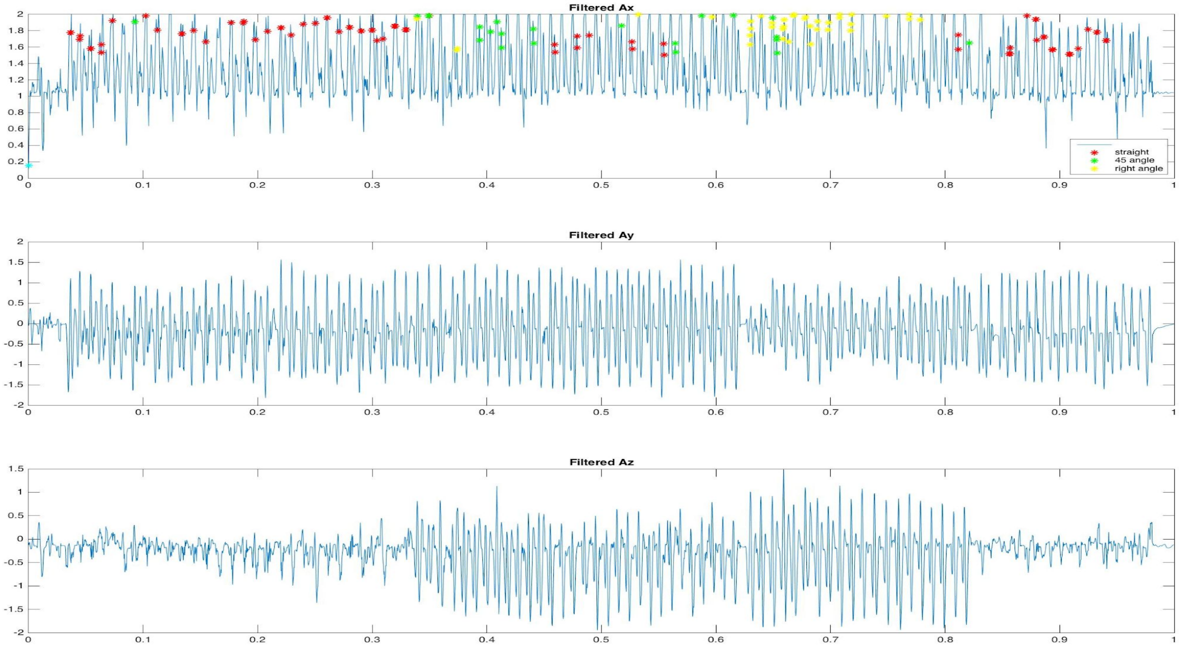


All components in kick tracker are connected together in PCB.
Battery is rechargeable.

User data consists of user's score in this play through.

Example Measurements

- Raw time series is averaged over a window of last 5 samples to reduce noise and peakiness
- Graph shown on right consists of 30 straight kicks, 30 kicks at 45°, and 20 kicks at 90°



App Developments

Deliverables:

Cleaned up menu interface

Implemented left and right foot swap

Additional Features:

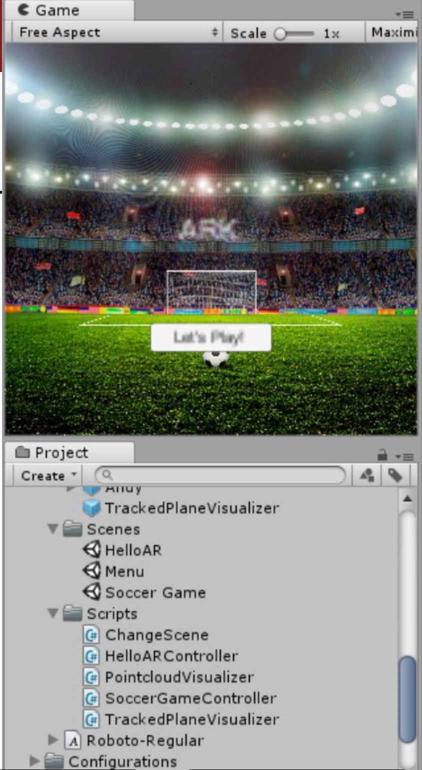
Can easily toggle hidden debugging features

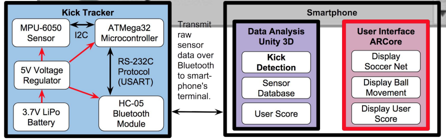
Added game timer

Added scoreboard

Can email your score

Locked screen rotation





Transmit raw sensor data over Bluetooth to smart-phone's terminal.

All components in kick tracker are connected together in PCB.
Battery is rechargeable.

User data consists of user's score in this play through.

Budget & Cost Analysis: Production Cost

Items Per Device	#	Bulk Cost at Bulk Quantity	Distributor
ATMega32	1	$4.80 @ 100	Digikey
HC-05 Bluetooth	1	$2.75 @ 1	Alibaba
MPU6050 Inertial Sensor	1	$4.07 @ 1000	Digikey
Voltage Regulator	1	$3.49 @ 100	Pololu
Battery	1	$3.25 @ 4	Amazon
PCB & Case	1	$6.00	Estimation
Starlight Headset	1	$13.99 @ 1	Amazon
Total:		**$38.05**	

1. Device is only $24.06 if user has their own headset.

2. We have approximately $100 left of our $500 total prototyping budget.

Demo

Thank You!

Questions?